



Artificial Intelligence 101 (1998) 251–265

**Artificial
Intelligence**

Research Note

Constraints, consistency and closurePeter Jeavons^{a,*}, David Cohen^a, Martin C. Cooper^b^a *Department of Computer Science, Royal Holloway, University of London, Egham, Surrey TW20 OEX, UK*^b *University of Toulouse, France*

Received 20 October 1996; received in revised form 17 February 1998

Abstract

Although the constraint satisfaction problem is NP-complete in general, a number of constraint classes have been identified for which some fixed level of local consistency is sufficient to ensure global consistency. In this paper we describe a simple algebraic property which characterises all possible constraint types for which strong k -consistency is sufficient to ensure global consistency, for each $k > 2$. We give a number of examples to illustrate the application of this result. © 1998 Elsevier Science B.V. All rights reserved.

Keywords: Constraint class; Global consistency; Local consistency; Tractability; Algebraic closure

1. Introduction

The constraint satisfaction problem provides a framework in which it is possible to express, in a natural way, many combinatorial problems encountered in artificial intelligence and elsewhere. The aim in a constraint satisfaction problem is to find an assignment of values to a given set of variables subject to constraints on the values which can be assigned simultaneously to certain specified subsets of variables.

The constraint satisfaction problem is known to be an NP-complete problem in general [22]. However, by imposing restrictions on the constraint interconnections [8,12,14,23,24], or on the form of the constraints [5,16,19,20,23,28,30], it is possible to obtain restricted versions of the problem that are tractable.

Our long term research aim is to determine all possible restrictions on the form of the constraints that ensure tractability. We call this research problem the characterisation of tractable constraints. It has already been solved in the special, but very important, case in which all the domains contain just two possible values. In this case the constraint

* Corresponding author. Email: pete@dcs.rhnc.ac.uk.

satisfaction problem is equivalent to the GENERALIZED SATISFIABILITY problem first described by Schaefer in [25]. Schaefer [25] established that for this problem there exist only three nontrivial forms of tractable constraints: those expressible using Horn clauses (or a dual version of Horn clauses), those expressible using binary constraints and those that are affine.

All of these classes of tractable constraints have now been generalised to obtain maximal tractable classes over larger domains:

- Horn clauses have been generalised to the ‘max-closed’ constraints over arbitrary ordered domains first identified in [19]. A further generalisation, to partially ordered domains, is described in Section 5 of [16].
- Binary Boolean constraints have been generalised to the ‘0/1/all’ constraints first identified in [5] (and independently in [20]). A further generalisation is described in Section 5.2 of [18].
- Affine constraints over a Boolean domain have been generalised to affine constraints over an arbitrary domain with a prime number of elements (not necessarily numerical elements) [16]. A further generalisation to domains of arbitrary size is described in Section 5.4 of [18].

We have recently developed a novel approach to the study of tractable constraint types which focuses on certain algebraic closure properties of constraints [16,17] (the precise definition of the form of closure property used in this approach is given below). This approach has established that any collection of tractable constraint types over a finite domain, D , must all be closed under a pointwise operation of order at most $|D|$ [17, 18]. This result transforms the search for new tractable constraint types into a search for possible algebraic closure properties.

An alternative approach to identifying some forms of tractable constraints has been developed by Dechter [6] and van Beek [27]. This work is based on the observation that problems for which ‘local consistency’ operations are sufficient to ensure ‘global consistency’ can be solved in polynomial time. One example of this is the 2-SATISFIABILITY problem, for which strong 3-consistency is sufficient to ensure global consistency [6]. Another example is the class of ‘row-convex’ constraints, introduced in [27], and extended in [28], for which strong 3-consistency is sufficient to ensure global consistency [28]. Another example, this time involving an infinite domain, is the class of ‘simple temporal constraints’ described in [7], for which strong 3-consistency is again sufficient to ensure global consistency [7].

The major contribution of this paper is the characterisation of all possible constraint types for which strong k -consistency guarantees global consistency, for each $k > 2$. We give necessary and sufficient algebraic conditions on collections of constraint types for local consistency to ensure global consistency, and hence we bring together two fundamentally different approaches to the study of tractability.

The paper is organised as follows. In Section 2 we give the basic definitions and describe a general form of algebraic closure condition for a set of relations. In Section 3 we define the ideas of local and global consistency and establish the connection between certain algebraic closure properties and the sufficiency of local consistency for global consistency, and in Section 4 we give applications and examples to illustrate the power of this result.

2. Definitions

2.1. The constraint satisfaction problem

The fundamental mathematical structure required to describe constraints, and constraint satisfaction problems, is the *relation*, which is defined as follows.

Definition 2.1. For any set D , and any natural number n , we denote the set of all n -tuples of elements of D by D^n . A subset of D^n is called an ' n -ary relation' over D .

For any tuple $t \in D^n$, and any i in the range 1 to n , we denote the value in the i th coordinate position of t by $t[i]$. The tuple t will be written in the form $\langle t[1], t[2], \dots, t[n] \rangle$, and the length of t will be denoted $|t|$.

We now define the 'constraint satisfaction problem' [21–23].

Definition 2.2. An instance of a *constraint satisfaction problem* consists of:

- a finite set of variables, V ;
- a finite domain of values, D ;
- a finite set of constraints $\{C_1, C_2, \dots, C_q\}$;
each constraint C_i is a pair (s_i, R_i) , where:
 - s_i is a tuple of variables of length m_i , called the 'constraint scope'; and
 - R_i is an m_i -ary relation over D , called the 'constraint relation'.

For each constraint, (s_i, R_i) , the tuples in R_i indicate the allowed combinations of simultaneous values for the variables in s_i . The length of s_i , and of the tuples in R_i , is called the 'arity' of the constraint. In particular, unary constraints specify the allowed values for a single variable, and binary constraints specify the allowed combinations of values for a pair of variables.

A *solution* to a constraint satisfaction problem instance is a function from the variables to the domain such that the image of each constraint scope is an element of the corresponding constraint relation. Deciding whether or not a given problem instance has a solution is NP-complete in general [22], even when the constraints are restricted to binary constraints. In this paper we shall consider how restricting the allowed constraint relations to some fixed subset of all the possible relations affects the complexity of this decision problem. We therefore make the following definition.

Definition 2.3. For any set of relations, Γ , C_Γ is defined to be the class of decision problems with:

Instance: A constraint satisfaction problem instance, \mathcal{P} , in which all constraint relations are elements of Γ or binary equality relations.

Question: Does \mathcal{P} have a solution?

If there is an algorithm which solves every problem instance in C_Γ in polynomial time, then we shall say that Γ is a 'tractable' set of relations.

Example 2.4. The binary disequality relation over a set D is defined as follows.

$$\neq_D = \{\langle d_1, d_2 \rangle \in D^2 \mid d_1 \neq d_2\}.$$

For any finite set D , the class of constraint satisfaction problem instances $C_{\{\neq_D\}}$ corresponds to the GRAPH COLOURABILITY problem [13] with $|D|$ colours. This problem is tractable when $|D| \leq 2$ and NP-complete when $|D| \geq 3$.

2.2. Operations on relations

In order to describe operations which can be carried out on constraint relations it is convenient to make use of the following standard operations from relational algebra [2].

Definition 2.5.

- Let R_1 be an n -ary relation over a domain D and let R_2 be an m -ary relation over D . The *Cartesian product* $R_1 \times R_2$ is defined to be the $(n + m)$ -ary relation

$$R_1 \times R_2 = \{\langle t[1], t[2], \dots, t[n + m] \rangle \mid (\langle t[1], t[2], \dots, t[n] \rangle \in R_1) \wedge (\langle t[n + 1], t[n + 2], \dots, t[n + m] \rangle \in R_2)\}.$$

- Let R be an n -ary relation over a domain D . Let $1 \leq i, j \leq n$. The *equality selection* $\sigma_{i=j}(R)$ is defined to be the n -ary relation

$$\sigma_{i=j}(R) = \{t \in R \mid t[i] = t[j]\}.$$

- Let t be an n -tuple and let (i_1, \dots, i_k) be a list of indices chosen from $\{1, 2, \dots, n\}$. The *projection* $\pi_{i_1, \dots, i_k}(t)$ is defined to be the k -tuple

$$\pi_{i_1, \dots, i_k}(t) = \langle t[i_1], \dots, t[i_k] \rangle.$$

Similarly, for any n -ary relation R , the *projection* $\pi_{i_1, \dots, i_k}(R)$ is defined to be the k -ary relation

$$\pi_{i_1, \dots, i_k}(R) = \{\langle t[i_1], \dots, t[i_k] \rangle \mid t \in R\}.$$

The combined effect of two constraints in a constraint satisfaction problem can be obtained by performing a relational *join* operation [2] on the two constraint relations [14]. The result of such a join operation can also be calculated by performing a sequence of Cartesian product, equality selection and projection operations on the constraint relations [3]. We therefore introduce the following notation.

Notation 2.6. The set of all relations which can be obtained from a given set of relations, Γ , using some sequence of Cartesian product, equality selection, and projection operations will be denoted Γ^+ .

2.3. Operations on tuples

Any operation on the elements of a set D can be extended to an operation on tuples over D by applying the operation to the values in each coordinate position separately.

Hence, any operation defined on the domain of a relation can be used to define an operation on the elements of that relation, as follows:

Definition 2.7. Let R be an n -ary relation over a domain D , and let $\varphi: D^k \rightarrow D$ be a k -ary operation on D .

For any collection of tuples, $t_1, t_2, \dots, t_k \in R$ (not necessarily all distinct), define the tuple $\varphi(t_1, t_2, \dots, t_k)$ as follows:

$$\varphi(t_1, t_2, \dots, t_k) = \langle \varphi(t_1[1], t_2[1], \dots, t_k[1]), \varphi(t_1[2], t_2[2], \dots, t_k[2]), \dots, \varphi(t_1[n], t_2[n], \dots, t_k[n]) \rangle.$$

Using this definition, we now define the following closure property of relations.

Definition 2.8. Let R be a relation over a domain D , and let $\varphi: D^k \rightarrow D$ be a k -ary operation on D .

R is said to be *closed under φ* if, for all $t_1, t_2, \dots, t_k \in R$ (not necessarily all distinct),

$$\varphi(t_1, t_2, \dots, t_k) \in R.$$

Example 2.9. Let μ be the ternary operation defined as follows:

$$\mu(x, y, z) = \begin{cases} y & \text{if } y = z; \\ x & \text{otherwise.} \end{cases}$$

The relation $\neq_{\{0,1\}}$ defined in Example 2.4 is closed under μ , since applying the μ operation to any three elements of $\neq_{\{0,1\}}$ yields an element of $\neq_{\{0,1\}}$. For example,

$$\mu(\langle 0, 1 \rangle, \langle 1, 0 \rangle, \langle 1, 0 \rangle) = \langle 1, 0 \rangle \in \neq_{\{0,1\}}.$$

The relation $\neq_{\{0,1,2\}}$, also defined in Example 2.4, is *not* closed under μ , since applying the μ operation to certain collections of 3 elements of $\neq_{\{0,1,2\}}$ yields a tuple which is not an element of $\neq_{\{0,1,2\}}$. For example,

$$\mu(\langle 1, 2 \rangle, \langle 0, 1 \rangle, \langle 2, 1 \rangle) = \langle 1, 1 \rangle \notin \neq_{\{0,1,2\}}.$$

The next lemma indicates that the property of being closed under some operation is preserved by each of the operations on relations described above.

Lemma 2.10. Let R_1 and R_2 be relations which are closed under φ , for some operation φ . The following relations are also closed under φ :

- (1) the Cartesian product, $R_1 \times R_2$;
- (2) any projection of R_1 or R_2 ;
- (3) any equality selection from R_1 or R_2 .

Proof. Follows immediately from the definitions. \square

We shall be particularly interested in operations known as ‘near unanimity operations’ [1,26], which are defined as follows.

Definition 2.11. An operation $\varphi: D^k \rightarrow D$, where $k \geq 3$, is called a ‘near unanimity operation’ if, for all $d, e \in D$,

$$\varphi(e, d, d, \dots, d) = \varphi(d, e, d, \dots, d) = \dots = \varphi(d, d, \dots, d, e) = d.$$

(In other words, whenever $k - 1$ arguments have the same value, then that value must be returned by the operation. In all other cases any value in D may be returned.)

A ternary near unanimity operation is called a ‘majority operation’ [26].

Example 2.12.

- On any domain D , the ternary operation μ , defined in Example 2.9, is a majority operation.
- On any totally ordered domain, the k -ary *median* operation η_k , which returns the median value of its k arguments, is a near unanimity operation, for any $k \geq 3$. In particular, the ternary median operation, η_3 , is a majority operation.
- On the domain $\{0, 1\}$, the k -ary *threshold* operation, $\theta_{k,m}$, which returns 1 if at least m of its k arguments are 1, and 0 otherwise, is a near unanimity operation for any $k \geq 3$ and $2 \leq m \leq k - 1$.

3. Consistency, decomposability and closure

The notion of ‘consistency’ has proved to be very useful in the analysis of constraint satisfaction problems [6,11,12,22]. To define this notion, we first introduce the idea of a ‘subproblem’ of a constraint satisfaction problem which is generated by a subset of the variables.

Definition 3.1. Let \mathcal{P} be a constraint satisfaction problem instance with set of variables V , domain D and constraints C .

For any subset V' of V , the *subproblem of \mathcal{P} generated by V'* , denoted $\mathcal{P}|_{V'}$, is the problem instance with set of variables V' , and domain D , where the constraints are obtained from the constraints of \mathcal{P} as follows: for each constraint (s, R) of \mathcal{P} , such that s contains elements of V' , choose $I = (i_1, i_2, \dots, i_r)$ to be a list of the indices of the elements of V' in s , and make $(\pi_I(s), \pi_I(R))$ a constraint of $\mathcal{P}|_{V'}$.

Definition 3.2. A constraint satisfaction problem instance \mathcal{P} is said to be *i -consistent* if for any subset V' containing $i - 1$ variables, and any variable v , any solution to $\mathcal{P}|_{V'}$ can be extended to a solution to $\mathcal{P}|_{V' \cup \{v\}}$.

If \mathcal{P} is j -consistent for $j = 2, 3, \dots, i$, then it is said to be *strong i -consistent*. The maximum value of i such that \mathcal{P} is strong i -consistent is called the *degree of consistency* of \mathcal{P} .

If \mathcal{P} is j -consistent for all j , then it is said to be *globally consistent*.

Any constraint satisfaction problem instance \mathcal{P} can be modified to obtain an i -consistent problem instance \mathcal{P}' without changing the set of solutions by solving all subproblems involving i variables, and then imposing constraints on all subsets of $i - 1$ variables that allow only these solutions. This procedure is called ‘establishing i -consistency’ [4].

For most constraints of arity n it is impossible to achieve the same constraint using constraints of smaller arity on the same variables. However, certain relations have the

property that they *can* be replaced by a collection of projections of smaller arity which together impose exactly the same constraint¹. To describe this idea precisely, we make the following definition.

Definition 3.3. An n -ary relation R over domain D is said to be r -decomposable if it contains all n -tuples t such that $\pi_I(t) \in \pi_I(R)$ for all lists of indices, I , from the set $\{1, 2, \dots, n\}$, with $|I| \leq r$.

Example 3.4. Any relation containing a single tuple is r -decomposable for all $r \geq 1$.

For example, let $D = \{0, 1\}$, and for any $n \geq 1$ set $U_n = \{t_n\}$, where $t_n = \langle 0, 0, \dots, 0 \rangle$ is the n -ary tuple of zeros. Note that

$$\pi_I(U_n) = \left\{ \overbrace{\langle 0, 0, \dots, 0 \rangle}^{|I|} \right\}$$

for any I , so t_n is the only n -tuple t such that $\pi_I(t) \in \pi_I(U_n)$ for all I . Hence, U_n is r -decomposable for all $r \geq 1$.

On the other hand, the *complement* of a relation containing a single tuple is generally *not* r -decomposable for all $r \geq 1$.

For example, let $D = \{0, 1\}$, and for any $n \geq 1$ set $T_n = D^n \setminus \{t_n\}$, where $t_n = \langle 0, 0, \dots, 0 \rangle$ is the n -ary tuple of zeros. Note that

$$\pi_I(t_n) = \left\{ \overbrace{\langle 0, 0, \dots, 0 \rangle}^{|I|} \right\} \in \pi_I(T_n)$$

for any I with $|I| \leq n - 1$, but $t_n \notin T_n$. Hence, T_n is *not* $(n - 1)$ -decomposable.

We now present the main result of this paper, which links algebraic properties of relations, decomposability, and the effectiveness of local consistency.

Theorem 3.5. For any set of relations Γ , over a finite set D , and any $r \geq 3$, the following conditions are equivalent:

- (1) Every R in Γ is closed under a near unanimity operation, ν , of arity r .
- (2) Every R in Γ^+ is $(r - 1)$ -decomposable.
- (3) For every \mathcal{P} in \mathbf{CR} , establishing strong r -consistency ensures global consistency.

Proof. (1) \Rightarrow (2) Let Γ be a set of relations such that every $R \in \Gamma$ is closed under the near unanimity operation ν of arity r , and let R be an element of Γ^+ of arity n . We shall prove by induction on n that R is $(r - 1)$ -decomposable.

For $n < r$ the result holds trivially, so assume that $n \geq r$ and the result holds for all smaller values of n . Let t be any n -tuple such that $\pi_I(t) \in \pi_I(R)$ for all lists of indices, I chosen from $\{1, 2, \dots, n\}$ with $|I| \leq r - 1$. We need to show that $t \in R$.

¹ This property of a relation corresponds to the notion of satisfying a join dependency, from relational database theory [14].

For $i = 1, 2, \dots, r$ consider the relation $R_i = \pi_{1,2,\dots,i-1,i+1,\dots,n}(R)$. By Lemma 2.10, each R_i is closed under ν . By the inductive hypothesis, applied to each R_i ,

$$\pi_{1,2,\dots,i-1,i+1,\dots,n}(t) \in R_i,$$

so for $i = 1, 2, \dots, r$ there is some tuple $t_i \in R$ which agrees with t at all coordinate positions except (possibly) i . But this implies that $\nu(t_1, t_2, \dots, t_r) = t$, and R is closed under ν , by Lemma 2.10, so $t \in R$, and the result follows.

(2) \Rightarrow (1) Let Γ be a set of relations over a finite set D such that every R in Γ^+ is $(r-1)$ -decomposable, and let Γ_{r-1}^+ be the (finite) set of all relations in Γ^+ with arity at most $r-1$.

Consider the problem instance $\mathcal{P} \in C_{\Gamma^+}$ with set of variables D' , domain D , and constraints defined as follows: for each $R \in \Gamma_{r-1}^+$, and for each sequence t_1, t_2, \dots, t_r of tuples from R , form a constraint (s, R) with $s = \langle v_1, v_2, \dots, v_n \rangle$ where n is the arity of R and $v_j = \langle t_1[j], t_2[j], \dots, t_r[j] \rangle$, for $j = 1, 2, \dots, n$.

This problem is the ‘indicator problem’ of order r for Γ_{r-1}^+ , as defined in [17]. It is shown in [17] (and it follows from the definitions above), that the solutions to \mathcal{P} are precisely the operations of arity r under which every $R \in \Gamma_{r-1}^+$ is closed. In particular, note that the solutions to \mathcal{P} include all the ‘projection’ operations, ϕ_i , for $i = 1, 2, \dots, r$, defined by $\phi_i(d_1, d_2, \dots, d_r) = d_i$ for all $d_1, d_2, \dots, d_r \in D$.

Let $W = \{w_1, w_2, \dots, w_m\}$ be the subset of variables of \mathcal{P} that are tuples containing at least $r-1$ identical values, and consider the relation R_W given by

$$R_W = \{ \langle \sigma(w_1), \sigma(w_2), \dots, \sigma(w_m) \rangle \mid \sigma \text{ is a solution to } \mathcal{P} \}.$$

This relation can be obtained by a sequence of Cartesian product, equality selection and projection operations from the constraint relations in \mathcal{P} , so it is an element of Γ^+ , and hence is $(r-1)$ -decomposable, by the choice of Γ .

Now consider the mapping ξ which assigns to each element w_i of W the repeated value in the tuple w_i , and set $t = \langle \xi(w_1), \xi(w_2), \dots, \xi(w_m) \rangle$. Note that, for any subset of W containing $r-1$ or fewer variables, there is some projection operation ϕ_i which agrees with ξ on these variables. Hence, for any list of indices I chosen from $\{1, 2, \dots, m\}$ with $|I| \leq r-1$ we have $\pi_I(t) \in \pi_I(R_W)$, so $t \in R_W$, by the decomposability of R_W .

But, by the definition of R_W , this implies that ξ can be extended to a solution to \mathcal{P} , and this extension must be a near-unanimity operation of arity r , by the choice of ξ . Hence, every $R \in \Gamma_{r-1}^+$ is closed under a near unanimity operation of arity r . By the decomposability of Γ^+ , this implies that every $R \in \Gamma^+$ is closed under this near unanimity operation, so the result follows.

(2) \Rightarrow (3) Let Γ be a set of relations over D such that every R in Γ^+ is $(r-1)$ -decomposable, let \mathcal{P}_0 be any element of C_Γ , and let \mathcal{P} be the problem instance obtained by establishing strong r -consistency in \mathcal{P}_0 .

Assume, for contradiction, that \mathcal{P} is not globally consistent. This means that there is some j such that \mathcal{P} is not j -consistent, which means that there is some subset $W = \{w_1, w_2, \dots, w_{j-1}\}$ of the variables of \mathcal{P} , and some variable v of \mathcal{P} , such that there is a solution, ξ , to $\mathcal{P}|_W$ which cannot be extended to a solution to $\mathcal{P}|_{W \cup \{v\}}$.

This implies that $\mathcal{P}|_{W \cup \{v\}}$ has at least one constraint. Let the constraints of $\mathcal{P}|_{W \cup \{v\}}$ be $(s_1, R_1), (s_2, R_2), \dots, (s_q, R_q)$. To obtain the desired contradiction, we shall construct a

new problem, $\mathcal{P}' \in \mathcal{C}_{\Gamma^+}$, which also has q constraints, with the same constraint relations, but with different constraint scopes.

We define the set of variables of \mathcal{P}' to be $\bigcup_{i=1}^q W_i \cup \{v'\}$, where W_1, W_2, \dots, W_q is a disjoint collection of sets each of cardinality $j-1$.

Now, for $i = 1, 2, \dots, q$, we define an injective mapping $f_i : W \rightarrow W_i$, and extend each f_i to v by setting $f_i(v) = v'$. The set of constraints of \mathcal{P}' is then defined as:

$$\{(f_1(s_1), R_1), (f_2(s_2), R_2), \dots, (f_q(s_q), R_q)\}$$

(where the result of applying the function f_i to a tuple is defined to be the tuple of values obtained by applying it to each coordinate position separately).

Now we define the $q * (j-1)$ -ary relation R as follows

$$R = \{(\sigma(f_1(w_1)), \dots, \sigma(f_1(w_{j-1})), \sigma(f_2(w_1)), \dots, \sigma(f_2(w_{j-1})), \dots, \sigma(f_q(w_1)), \dots, \sigma(f_q(w_{j-1}))) \mid \sigma \text{ is a solution to } \mathcal{P}'\}.$$

Note that $R \in \Gamma^+$. However, we will now show that R is *not* $(r-1)$ -decomposable.

To do this we consider the $q * (j-1)$ -tuple t , defined by

$$t = \langle \xi(w_1), \dots, \xi(w_{j-1}), \xi(w_1), \dots, \xi(w_{j-1}), \dots, \xi(w_1), \dots, \xi(w_{j-1}) \rangle.$$

For any list of indices I chosen from $\{1, 2, \dots, q * (j-1)\}$, with $|I| \leq r-1$, we claim that $\pi_I(t) \in \pi_I(R)$. To establish this claim we recall that \mathcal{P} is strong r -consistent, so for any subset W' of W with $|W'| \leq r-1$, the restriction of ξ to W' can be extended to a solution $\tilde{\xi}$ to $\mathcal{P}|_{W' \cup \{v\}}$. In particular, this holds for $W' = \{w_{(i \bmod (j-1)) + 1} \mid i \in I\}$. Furthermore, for any solution $\tilde{\xi}$ to $\mathcal{P}|_{W' \cup \{v\}}$ we can construct a corresponding solution, $\tilde{\xi}'$, to \mathcal{P}' , such that $\tilde{\xi}'(f_i(w)) = \tilde{\xi}(w)$ for all $w \in W'$ and for all $i \in \{1, 2, \dots, q\}$, by the construction of \mathcal{P}' . Hence t agrees with some element of R in the set of positions indexed by I , which establishes the claim.

On the other hand, we note that $t \notin R$, since ξ cannot be extended to a solution to $\mathcal{P}|_{W \cup \{v\}}$, by the choice of ξ .

Hence R is not $(r-1)$ -decomposable, which contradicts the choice of Γ , and the result follows.

(3) \Rightarrow (2) Let Γ be a set of relations over a set D such that every \mathcal{P} in \mathcal{C}_{Γ} is globally consistent after establishing strong r -consistency.

Now let R be any relation in Γ^+ . By the definition of Γ^+ , we can construct some \mathcal{P} in \mathcal{C}_{Γ} with some set of variables $V \supseteq \{v_1, v_2, \dots, v_n\}$ such that

$$R = \{(\sigma(v_1), \sigma(v_2), \dots, \sigma(v_n)) \mid \sigma \text{ is a solution to } \mathcal{P}\}.$$

Consider the modified problem instance \mathcal{P}' which is obtained from \mathcal{P} by adding an extra set of variables $V' = \{v'_1, v'_2, \dots, v'_n\}$, then adding a binary equality constraint on each pair v_i, v'_i , and finally establishing strong r -consistency. By the choice of Γ , \mathcal{P}' is globally consistent, so any solution to $\mathcal{P}'|_{V'}$ can be extended to a complete solution to \mathcal{P}' .

However, the only constraints in $\mathcal{P}'|_{V'}$ are introduced to enforce strong r -consistency, so they have arity at most $r-1$. Hence, any tuple t such that $\pi_I(t) \in \pi_I(R)$ for all lists of indices I chosen from $\{1, 2, \dots, n\}$ with $|I| \leq r-1$ must correspond to a solution to $\mathcal{P}'|_{V'}$. Note that, because of the equality constraints in \mathcal{P}' , for each solution σ' to $\mathcal{P}'|_{V'}$ there is

a corresponding solution σ to \mathcal{P} with $\sigma(v_i) = \sigma'(v'_i)$ for $i = 1, 2, \dots, n$. Hence, $t \in R$ and so R is $(r - 1)$ -decomposable. \square

We note that this theorem can also be obtained as an application of a result in universal algebra which was established by Baker and Pixley in 1975 [1], but this requires the use of more extensive algebraic terminology than we wish to introduce here. A related result was obtained by Feder and Vardi in [10], but they do not make any explicit links with the well-established ideas of consistency in constraint satisfaction problems, which is our primary motivation here.

A number of efficient algorithms have been developed for establishing strong r -consistency, for any fixed value of r [4]. Using the equivalence in Theorem 3.5 in one direction we can show that for some problem classes these algorithms are sufficient to provide a complete solution, which gives the following result.

Corollary 3.6. *For any set of relations Γ over a finite domain, if every $R \in \Gamma$ is closed under a near-unanimity operation, v , then C_Γ is tractable.*

Proof. Let every $R \in \Gamma$ be closed under a near-unanimity operation v , of arity r , and let \mathcal{P} be any problem instance in C_Γ . For any fixed value of r , \mathcal{P} can be made strong r -consistent in polynomial time [4]. By Theorem 3.5 this ensures global consistency, and hence a solution can be found without backtracking. \square

The special case of this result for majority operations was given in [18].

Using the equivalence in Theorem 3.5 in the opposite direction we can show that there are efficient techniques to determine, for a given collection of constraint types, whether r -consistency is sufficient to ensure global consistency.

Corollary 3.7. *For any finite domain D , and any $r \geq 3$, there is a polynomial-time algorithm which determines, for a set of relations Γ over D , whether strong r -consistency is sufficient to ensure global consistency in C_Γ .*

Proof. By Theorem 3.5, we simply need to establish whether or not there is a near-unanimity operation of arity r , under which every $R \in \Gamma$ is closed. For each D and r there are a constant number of such operations, so this can be established in polynomial time (in the size of Γ). \square

Finally, what can be said about sets of relations for which 2-consistency is sufficient to ensure global consistency in all problem instances? The proof of Theorem 3.5 establishes that a set of relations has this property if and only if each relation is 1-decomposable. This means that each relation must be a Cartesian product, and hence each constraint with such a relation is equivalent to a collection of unary constraints, so this case is rather trivial.

The results of this section concern arbitrary problem instances containing constraint relations from a specified set of relations. In contrast, there are a number of earlier results linking local consistency and global consistency [6,29] which concern restricted problem instances with particular properties. For example, it was shown in [6] that any constraint

satisfaction problem instance over a domain D , with constraints of arity at most s , that is strong $((|D|(s - 1) + 1)$ -consistent is globally consistent. This result was strengthened in [29] by defining a property of a relation called ‘ m -tightness’, and showing that any constraint satisfaction problem instance, with m -tight constraints of arity at most s , that is strong $((m + 1)(s - 1) + 1)$ -consistent is globally consistent. Note that in both of these results the degree of consistency required is not fixed, but depends on the maximum arity of the constraints. For an arbitrary problem instance, establishing k -consistency may increase the maximum constraint arity to $k - 1$, and hence increase the degree of consistency required to apply these results. This means that these earlier results cannot be used directly to establish the tractability of a given set of relations for arbitrary problem instances, except in special cases.²

4. Applications and examples

4.1. Row-convex constraints

The class of binary row-convex constraint relations [27] can be defined as follows:

Definition 4.1. A binary relation, R , over an ordered set D is *row-convex* if, for all $d, d_1, d_2, d_3 \in D$ such that $d_1 < d_2 < d_3$ the following implication holds:

$$\langle d, d_1 \rangle \in R \text{ and } \langle d, d_3 \rangle \in R \Rightarrow \langle d, d_2 \rangle \in R.$$

It has been shown that all problems involving only binary row-convex constraints which are strong 3-consistent are also globally consistent.

Theorem 4.2 [27,28]. *If every relation in Γ is row-convex, then any strong 3-consistent problem instance in C_Γ is globally consistent.*

However, the fact that every relation in some set of relations Γ is row-convex is *not* in general a sufficient condition to ensure that C_Γ is tractable. This is because establishing 3-consistency may introduce new constraint relations into a problem instance that are not row-convex [28]. By combining Theorem 3.5 with Theorem 4.2, we can now identify exactly when this problem can and cannot occur.

Corollary 4.3. *For any set of relations Γ over a finite domain, if, after establishing strong 3-consistency, every constraint relation in every problem instance in C_Γ is row-convex, then every relation in Γ is closed under a majority operation, v .*

Using this result we now give some examples of maximal families of row-convex constraints which do guarantee tractability.

² In the special cases when $s = 2$ and either $|D| = 2$ or $m = 1$ these earlier results do show that 3-consistency is sufficient to ensure global consistency in arbitrary problem instances. These cases correspond to the tractable sets of binary relations described in Examples 4.4 and 4.5, respectively.

Example 4.4. All binary relations over a domain with two elements are row-convex [28].

For any domain with just two elements, there is a unique majority operation, μ , as defined in Example 2.9. It is straightforward to verify that all binary relations over a domain with just two elements are closed under μ [16].

Hence, any class of problems involving binary constraints over a domain with two elements is tractable, and 3-consistency is sufficient to ensure global consistency.

Example 4.5. The binary 0/1/all relations introduced in [5] are row-convex.

It was shown in [16] that these are precisely the binary relations that are closed under the majority operation μ , defined in Example 2.9.

Hence any class of problems involving 0/1/all constraints is tractable, and 3-consistency is sufficient to ensure global consistency.

Example 4.6. The *implicational* relations described in [20] are row-convex (in the extended sense defined in [28]).

Binary implicational relations correspond precisely to the 0/1/all relations discussed in Example 4.5. Furthermore, implicational relations of arbitrary arity can always be replaced by an equivalent collection of binary constraints on the same variables [20], and hence they are 2-decomposable. In fact, it can be shown that implicational relations are precisely the set of all relations closed under the majority operation μ , defined in Example 2.9.

Hence, any class of problems involving implicational constraints is tractable, and 3-consistency is sufficient to ensure global consistency.

Example 4.7. As a final example of a class of row-convex relations which is tractable we consider the class of relations over an ordered domain $D = \{d_1, d_2, \dots\}$, with $d_1 < d_2 < \dots$, which is closed under the ternary median operation, η_3 , defined in Example 2.12.

To illustrate the form of relations closed under this operation, we represent a binary relation by a 0–1 matrix M in the standard way, by setting $M_{ij} = 1$ if the relation contains the pair $\langle d_i, d_j \rangle$, and 0 otherwise. The following relations are closed under η_3 :

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

In fact, the *binary* relations that are closed under η_3 can be described very simply: the pattern of 1's in the matrix representation (after removing rows and columns containing

only 0's) is connected along each row, along each column, and forms a connected two-dimensional region (where some of the connections may be diagonal).

This class of binary relations was introduced and shown to be tractable in [9], where they are called 'connected row-convex' relations. This class properly includes the 'monotone' relations, identified and shown to be tractable by Montanari in [23].

4.2. Other tractable constraint families

The next example shows that there are sets of relations which are *not* row-convex for which 3-consistency is still sufficient to ensure global consistency.

Example 4.8. Consider the relation R over the set $D = \{a, b, c, d\}$ defined by

$$R = \{\langle a, a \rangle, \langle a, b \rangle, \langle a, c \rangle, \langle a, d \rangle, \langle b, a \rangle, \langle b, b \rangle, \langle c, a \rangle, \langle c, c \rangle, \langle d, a \rangle, \langle d, d \rangle\}.$$

It was pointed out in [19] that this relation is not row-convex under any ordering of D . However, it is closed under the majority operation which returns the value a whenever all its arguments are distinct. Hence, by Theorem 3.5, if $\Gamma = \{R\}$ then 3-consistency is sufficient to ensure global consistency in C_Γ .

Furthermore, any other relations which are closed under the same majority operation can be added to Γ without losing this property. For example, all relations (of any arity) that contain only two tuples can be added to Γ to obtain a larger tractable set of relations.

4.3. Beyond 3-consistency

In all of the examples so far we have found that strong 3-consistency is sufficient to ensure global consistency. The next example shows that arbitrarily high levels of consistency are sometimes necessary.

Example 4.9. Recall the n -ary relation T_n over the set $\{0, 1\}$, defined in Example 3.4, which contains all tuples except $\langle 0, 0, \dots, 0 \rangle$, and choose some $n \geq 3$.

Note that T_n is closed under the $(n+1)$ -ary threshold operation $\theta_{n+1,2}$, so by Theorem 3.5, $(n+1)$ -consistency ensures global consistency for any problem instance in C_Γ .

To show that this is the minimal level of consistency which is sufficient, we simply note that T_n is *not* $(n-1)$ -decomposable, as shown in Example 3.4. Hence, by Theorem 3.5, T_n is not closed under any n -ary near unanimity operation, and n -consistency is *not* sufficient to ensure global consistency for every problem instance in C_Γ .

For example, consider the problem instance, \mathcal{P} in C_Γ , with variables $V = \{v_1, v_2, \dots, v_n, v'\}$ and constraints $C = \{C_0, C_1\}$ where:

$$C_0 = (\langle v_1, v_2, \dots, v_n \rangle, T_n), \quad C_1 = (\langle v_n, v' \rangle, \{\langle 0, 0 \rangle, \langle 1, 1 \rangle\}).$$

This problem instance is n -consistent, but not globally consistent, because the solution to the subproblem generated by $v_1, v_2, \dots, v_{n-1}, v'$ which assigns 0 to each variable cannot be extended to a complete solution to \mathcal{P} .

4.4. Infinite domains

If we extend the definition of a constraint satisfaction problem to allow infinite domains, and also allow instances to have (possibly) infinite sets of variables, and infinite sets of constraints, then Theorem 3.5 remains valid, and the proof is unchanged. This means that we can use this result to identify tractable constraint types over infinite domains, as the next example illustrates.

Example 4.10. One important class of constraint satisfaction problems are those involving restrictions on the timing of events or processes. These are known as *temporal problems* and they are described in [7].

We shall focus on a restricted class of temporal problems, which are referred to in [7] as ‘simple temporal problems’ (STP). In these problems, the variables must be assigned values from some infinite, densely ordered domain, which represents time. Constraints can be specified on pairs of variables in order to restrict the possible separations between their values. These constraints can be written as inequalities, for example $x_i - x_j \leq d$.

It is shown in [7] that for simple temporal problems 3-consistency is sufficient to ensure global consistency. This result can be obtained from Theorem 3.5 by noting that the constraints in such problems are all closed under the median operation, η_3 , defined in Example 2.12.

5. Conclusion

It was shown in [18] that *any* tractable set of relations must all be closed under an algebraic operation with certain restricted properties. The present paper has investigated a particular special form of tractable relations: relations which are tractable because some fixed level of local consistency is sufficient to ensure global consistency in all possible problem instances. We have described the algebraic conditions which characterise sets of relations with this property, and given a number of examples.

This result demonstrates once again the effectiveness of the algebraic approach to the classification of constraints which was developed in [16,17] and summarised in [18].

Acknowledgements

This research was partially supported by EPSRC research grant GR/L09936 and by the British Council.

References

- [1] K.A. Baker and A.F. Pixley, Polynomial interpolation and the Chinese remainder theorem, *Math. Z.* 143 (1975) 165–174.
- [2] E.F. Codd, A relation model of data for large shared databanks, *Comm. ACM* 13 (6) (1970) 377–387.

- [3] D.A. Cohen, M. Gyssens and P.G. Jeavons, Derivation of constraints and database relation, in: *Proceedings 2nd International Conference on Constraint Programming—CP'96*, Boston (August 1996), *Lecture Notes in Comput. Sci.*, Vol. 1118, Springer, Berlin, 1996, pp. 134–148.
- [4] M.C. Cooper, An optimal k -consistency algorithm, *Artificial Intelligence* 41 (1989) 89–95.
- [5] M.C. Cooper, D.A. Cohen and P.G. Jeavons, Characterising tractable constraints, *Artificial Intelligence* 65 (1994) 347–361.
- [6] R. Dechter, From local to global consistency, *Artificial Intelligence* 55 (1) (1992) 87–107.
- [7] R. Dechter, I. Meiri and J. Pearl, Temporal constraint networks, *Artificial Intelligence* 49 (1991) 61–95.
- [8] R. Dechter and J. Pearl, Network-based heuristics for constraint satisfaction problems, *Artificial Intelligence* 34 (1) (1988) 1–38.
- [9] Y. Deville, O. Baretté and P. van Hentenryck, Constraint satisfaction over connected row convex constraints, in: *Proceedings IJCAI'97*, Nagoya, Japan, 1997, pp. 405–411.
- [10] T. Feder and M.Y. Vardi, Monotone monadic SNP and constraint satisfaction, in: *Proceedings 25th ACM Symposium on the Theory of Computing (STOC)*, 1993, pp. 612–622.
- [11] E.C. Freuder, A sufficient condition for backtrack-free search, *J. ACM* 29 (1) (1982) 24–32.
- [12] E.C. Freuder, A sufficient condition for backtrack-bounded search, *J. ACM* 32 (1985) 755–761.
- [13] M. Garey and D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Freeman, San Francisco, CA, 1979.
- [14] M. Gyssens, P.G. Jeavons and D.A. Cohen, Decomposing constraint satisfaction problems using database techniques, *Artificial Intelligence* 66 (1) (1994) 57–89.
- [15] P.G. Jeavons and D.A. Cohen, An algebraic characterization of tractable constraints, in: *Computing and Combinatorics. First International Conference COCOON'95*, Xi'an, China (1995), *Lecture Notes in Comput. Sci.*, Vol. 959, Springer, Berlin, 1995.
- [16] P.G. Jeavons, D.A. Cohen and M. Gyssens, A unifying framework for tractable constraints, in: *Proceedings 1st International Conference on Constraint Programming—CP'95*, Cassis, France (September 1995), *Lecture Notes in Comput. Sci.*, Vol. 976, Springer, Berlin, 1995.
- [17] P.G. Jeavons, D.A. Cohen and M. Gyssens, A test for tractability, in: *Proceedings 2nd International Conference on Constraint Programming—CP'96*, Boston (August 1996), *Lecture Notes in Comput. Sci.*, Vol. 1118, Springer, Berlin, 1996, pp. 267–281.
- [18] P.G. Jeavons, D.A. Cohen and M. Gyssens, Closure properties of constraints, *J. ACM* 44 (1997) 527–548.
- [19] P.G. Jeavons and M.C. Cooper, Tractable constraints on ordered domains, *Artificial Intelligence* 79 (2) (1995) 327–339.
- [20] L. Kirousis, Fast parallel constraint satisfaction, *Artificial Intelligence* 64 (1993) 147–160.
- [21] P.B. Ladkin and R.D. Maddux, On binary constraint problems, *J. ACM* 41 (1994) 435–469.
- [22] A.K. Mackworth, Consistency in networks of relations, *Artificial Intelligence* 8 (1977) 99–118.
- [23] U. Montanari, Networks of constraints: fundamental properties and applications to picture processing, *Inform. Sci.* 7 (1974) 95–132.
- [24] U. Montanari and F. Rossi, Constraint relaxation may be perfect, *Artificial Intelligence* 48 (1991) 143–170.
- [25] T.J. Schaefer, The complexity of satisfiability problems, in: *Proceedings 10th ACM Symposium on Theory of Computing (STOC)*, 1978, pp. 216–226.
- [26] A. Szendrei, *Clones in Universal Algebra*, *Seminaires de Mathematiques Superieures*, Vol. 99, University of Montreal, Que., 1986.
- [27] P. van Beek, On the minimality and decomposability of row-convex constraint networks, in: *Proceedings AAAI-92*, San Jose, CA, 1992, pp. 447–452.
- [28] P. van Beek and R. Dechter, On the minimality and decomposability of row-convex constraint networks, *J. ACM* 42 (1995) 543–561.
- [29] P. van Beek and R. Dechter, Constraint tightness and looseness versus local and global consistency, *J. ACM* 44 (1997) 549–566.
- [30] P. van Hentenryck, Y. Deville and C.-M. Teng, A generic arc-consistency algorithm and its specializations, *Artificial Intelligence* 57 (1992) 291–321.